TLe

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/673,651	09/29/2003	Shyh-Kwei Chen	YOR920030265US1	2352

28211 7590 02/09/2007

FREDERICK W. GIBB, III
GIBB INTELLECTUAL PROPERTY LAW FIRM, LLC
2568-A RIVA ROAD
SUITE 304
ANNAPOLIS, MD 21401

EXAMINER	
COLAN, GIOVANNA B	
ART UNIT	**PAPER NUMBER**
2162	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/09/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/673,651	CHEN ET AL.	
	Examiner	Art Unit	
	Giovanna Colan	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 November 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,4,8,9,11,15,16,18,22,23,25,29,30,35-37,43,44,46 and 50-57* is/are pending in the application.

4a) Of the above claim(s) *5,12,19,26,33,39 and 48* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-4,6-11,13-18,20-25,27-32,34-38,40-47 and 49-57* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *09/11/2006*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is issued in response to applicant filed request for continued examination (RCE) on 11/06/06.

2. Claims 1, 4, 8 – 9,11, 15 – 16, 18, 22 – 23, 25, 29 – 30, 35 – 37, 43 – 44, 46, and 50 have been amended. Claims 51 – 57 were added. Claims 5, 12, 19, 26, 33, 39, and 48 were canceled.

3. Claims 1 – 4, 6 – 11, 13 – 18, 20 – 25, 27 – 32, 34 – 38, 40 – 47, and 49 – 57 are pending in this application.

4. Applicant's arguments with respect to amended claims 1, 4, 8 – 9,11, 15 – 16, 18, 22 – 23, 25, 29 – 30, 35 – 37, 43 – 44, 46, and 50 have been considered but are moot in view of the new ground(s) of rejection.

Information Disclosure Statement

The information disclosure statement (IDS) submitted on 09/11/2006 .The submission is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Continued Examination Under 37 CFR 1.114

5. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this

application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 10/11/2006 has been entered.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

8. Claims 1 – 4, 6, 8 – 11, 13, 15 – 18, 20, 22 – 25, 27, 29 – 32, 34, 36 – 38, 40 – 41, 43 – 47, and 50 – 57 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Reiner et al. (Reiner hereinafter) (US Patent No. 6,289,334 B1, issued: September 11, 2001) in view of Koskas (US Patent No. 6,711,563 B1, filed March 5, 2001).

Regarding Claim 1, Reiner discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals (Col. 10, lines 56 – 61, Reiner), said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, ... **generates multiple subqueries** ..., Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner).

Reiner also discloses: bitmap (Col. 63, lines 36 – 38, Reiner). However, Reiner does not explicitly disclose specific bit positions in bit map vectors. On the other hand, Koskas discloses specific bit positions in bit map vectors (Col. 11, lines 3 – 10, Koskas). It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the Koskas's teachings to the system Reiner. Skilled artisan would have been motivated to do so, as suggested by Koskas (Col. 4, lines 4 – 8, Koskas), to allocate respective identifiers to data graphs. In addition, both of the references (Reiner and Koskas) teach features that are directed to analogous art and they are directed to the same field of endeavor, such as, databases management systems, queries, and bitmap. This close relation between both of the references highly suggests an expectation of success.

Furthermore, the combination of Reiner in view of Koskas discloses:

determining predicate intervals, wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner); and

inserting each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, ... **appending** a predicate for matching records in the corresponding table partition ..., Reiner; and Col. 11, lines 3 – 10, Koskas) such that said specified ranges of data values of said predicate intervals are aligned with said predetermined ranges of said data values of said construct intervals (Col. 9, lines 36 – 39, Reiner[1]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 2, the combination of Reiner in view of Koskas discloses a method, wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals, and

wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), and

wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner).

[1] Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

Regarding Claim 3, the combination of Reiner in view of Koskas discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner[2]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 4, the combination of Reiner in view of Koskas discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner[3]) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner[4]).

Regarding Claim 6, the combination of Reiner in view of Koskas discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner[5]).

---

[2] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.
[3] Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.
[4] The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"
[5] Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

Regarding Claim 8, the combination of Reiner in view of Koskas discloses a method, wherein said inserting further comprises inserting said predicate intervals such that said virtual construct intervals holds multiples predicate intervals (Col. 2, lines 66 – 67, Reiner), wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 9, the combination of Reiner in view of Koskas discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner) and correspond to specific bit positions in bit map vectors (Col. 63, lines 36 – 38, Reiner; and Col. 11, lines 3 – 10, Koskas);

determining predicate intervals, wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner); and

inserting each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, ... appending a predicate for matching records in the corresponding table partition ..., Reiner; and Col. 11, lines 3 – 10, Koskas) such that said specified ranges of data values

of said predicate intervals are aligned with said predetermined ranges of said data values of said construct intervals (Col. 9, lines 36 – 39, Reiner[6]; and Col. 11, lines 3 – 10, Koskas),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner),

wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner),

wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), and

wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner).

Regarding Claim 10, the combination of Reiner in view of Koskas discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner[7]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 11, the combination of Reiner in view of Koskas discloses a method, wherein said process of defining said groups of virtual construct intervals

---

[6] Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

[7] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner[8]) and ending said virtual construct intervals in said group of virtual construct intervals at different data values (Col. 9, lines 17 – 28, Reiner[9]).

Regarding Claim 13, the combination of Reiner in view of Koskas discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner[10]).

Regarding Claim 15, the combination of Reiner in view of Koskas discloses a method, wherein said defining process only defines virtual construct intervals that are between the minimum and maximum possible data values of said predicate intervals (Col. 9 – 10 and 13, lines 67 and 1 – 2, and 43 – 45; respectively, Reiner).

Regarding Claim 16, the combination of Reiner in view of Koskas discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

---

[8] Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

[9] The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

[10] Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner) and correspond to specific bit positions in bit map vectors (Col. 63, lines 36 – 38, Reiner; and Col. 11, lines 3 – 10, Koskas);

determining predicate intervals, wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner); and

inserting each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, ... appending a predicate for matching records in the corresponding table partition ..., Reiner; and Col. 11, lines 3 – 10, Koskas) such that said specified ranges of data values of said predicate intervals are aligned with said predetermined ranges of said data values of said construct intervals (Col. 9, lines 36 – 39, Reiner[11]; and Col. 11, lines 3 – 10, Koskas),

wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner), and

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner).

Regarding Claim 17, the combination of Reiner in view of Koskas discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner[12]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 18, the combination of Reiner in view of Koskas discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at a same data value (Col. 10, lines 50 – 55, Reiner[13]) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner[14]).

Regarding Claim 20, the combination of Reiner in view of Koskas discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner[15]).

---

[11] Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.
[12] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.
[13] Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.
[14] The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2".
[15] Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

Regarding Claim 22, the combination of Reiner in view of Koskas discloses a method, wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner) and wherein said defining process only defines virtual construct intervals that are between the minimum and maximum possible data values of said predicate intervals (Col. 9 – 10 and 13, lines 67 and 1 – 2, and 43 – 45; respectively, Reiner).

Regarding Claim 23, the combination of Reiner in view of Koskas discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner) and correspond to specific bit positions in bit map vectors (Col. 63, lines 36 – 38, Reiner; and Col. 11, lines 3 – 10, Koskas);

determining predicate intervals, wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner); and

inserting each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28,

... appending a predicate for matching records in the corresponding table partition ..., Reiner; and Col. 11, lines 3 – 10, Koskas) such that said specified ranges of data values of said predicate intervals are aligned with said predetermined ranges of said data values of said construct intervals (Col. 9, lines 36 – 39, Reiner[16]; and Col. 11, lines 3 – 10, Koskas),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), and

wherein said defining process only defines virtual construct intervals that are between the minimum and maximum possible data values of said predicate intervals (Col. 9 – 10 and 13, lines 67 and 1 – 2, and 43 – 45; respectively, Reiner) .

Regarding Claim 24, the combination of Reiner in view of Koskas discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner[17]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 25, the combination of Reiner in view of Koskas discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals

[16] Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.
[17] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

at a same data value (Col. 10, lines 50 – 55, Reiner[18]) and ending said virtual construct intervals in said group of virtual construct intervals at different data values (Col. 9, lines 17 – 28, Reiner[19]).

Regarding Claim 27, the combination of Reiner in view of Koskas discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner[20]).

Regarding Claim 29, the combination of Reiner in view of Koskas discloses a method, wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner) and wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

---

[18] Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

[19] The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

[20] Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

Regarding Claim 30, the combination of Reiner in view of Koskas discloses a service adapted to maintains and use a query index, wherein queries within said query index have predicate intervals, said service:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner) and correspond to specific bit positions in bit map vectors (Col. 63, lines 36 – 38, Reiner; and Col. 11, lines 3 – 10, Koskas);

determining predicate intervals, wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner); and

inserting each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, ... appending a predicate for matching records in the corresponding table partition ..., Reiner; and Col. 11, lines 3 – 10, Koskas) such that said specified ranges of data values of said predicate intervals are aligned with said predetermined ranges of said data values of said construct intervals (Col. 9, lines 36 – 39, Reiner[21]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 31, the combination of Reiner in view of Koskas discloses a service, further comprising maintaining locations of said predicate intervals within said

groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner[22]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 32, the combination of Reiner in view of Koskas discloses a service, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner[23]) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner[24]).

Regarding Claim 34, the combination of Reiner in view of Koskas discloses a service, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner[25]).

---

[21] Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.
[22] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.
[23] Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.
[24] The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"
[25] Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

Regarding Claim 36, the combination of Reiner in view of Koskas discloses a service, said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner) and wherein said defining process only defines virtual construct intervals that are between the minimum and maximum possible data values of said predicate intervals (Col. 9 – 10 and 13, lines 67 and 1 – 2, and 43 – 45; respectively, Reiner).

Regarding Claim 37, the combination of Reiner in view of Koskas discloses a system for maintaining and using a query index, wherein queries within said query index have predicate intervals, said system comprising:

a plurality of bitmap vectors which define groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner) and correspond to specific bit positions in bit map vectors (Col. 63, lines 36 – 38, Reiner; and Col. 11, lines 3 – 10, Koskas);

a predicate insertion handler adapted to insert each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner) wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner),

wherein said predicate intervals are inserted such that said specified ranges of data values of said predicate intervals are aligned with said predetermined ranges of said data values of said virtual construct intervals.

Regarding Claim 38, the combination of Reiner in view of Koskas discloses a system, further comprising a predicate ID bitmap vector adapted to maintain locations of said predicate intervals within said groups of virtual construct intervals (Col. 63, lines 36 – 38, Reiner[26]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 40, the combination of Reiner in view of Koskas discloses a system, wherein all of said groups of said virtual construct intervals have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 41, the combination of Reiner in view of Koskas discloses a system, wherein said predicate insertion handler inserts said predicate intervals into the same sized virtual construct intervals (Col. 13, lines 55 – 57, Reiner[27]).

Regarding Claim 43, the combination of Reiner in view of Koskas discloses a system, wherein said groups of virtual construct intervals have uniform lengths (Col. 25,

[26] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

lines 44 – 47, Reiner), wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner) and wherein said defining process only defines virtual construct intervals that are between the minimum and maximum possible data values of said predicate intervals (Col. 9 – 10 and 13, lines 67 and 1 – 2, and 43 – 45; respectively, Reiner).

Regarding Claim 44, the combination of Reiner in view of Koskas discloses a program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner), wherein said virtual construct interval represent predetermined ranges of data values (Col. 26, lines 64 – 66, Reiner) and correspond to specific bit positions in bit map vectors (Col. 63, lines 36 – 38, Reiner; and Col. 11, lines 3 – 10, Koskas);

determining predicate intervals, wherein said predicate intervals represent specified ranges of data values from at least one of subscriptions, queries and rules (Col. 26, lines 64 – 66, Reiner); and

inserting each of said predicate intervals into said bit map positions of at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28,

---

[27] Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting

... appending a predicate for matching records in the corresponding table partition ..., Reiner; and Col. 11, lines 3 – 10, Koskas) such that said specified ranges of data values of said predicate intervals are aligned with said predetermined ranges of said data values of said construct intervals (Col. 9, lines 36 – 39, Reiner[28]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 45, the combination of Reiner in view of Koskas discloses a program storage device, wherein said method further comprises maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner[29]; and Col. 11, lines 3 – 10, Koskas).

Regarding Claim 46, the combination of Reiner in view of Koskas discloses a program storage device, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner[30]) and

---

predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

[28] Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

[29] Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

[30] Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner[31]).

Regarding Claim 47, the combination of Reiner in view of Koskas discloses a program storage device, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 50, the combination of Reiner in view of Koskas discloses a program storage device, wherein said inserting further comprises inserting said predicate intervals such that each of said virtual construct intervals holds multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner) and wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 51, 52, 53, 54, 55, 56, and 57, the combination of Reiner in view of Koskas discloses a program storage device, wherein said predetermined range

[31] The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

intervals of at least some of said virtual construct intervals between and within said groups of construct intervals overlap (Col. 9, lines 1 – 7, Reiner).

9. Claims 7, 14, 21, 28, 35, 42, and 49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Reiner et al. (Reiner hereinafter) (US Patent No. 6,289,334 B1, issued: September 11, 2001), in view of Koskas (US Patent No. 6,711,563 B1, filed March 5, 2001), and further in view of Jagadish et al. (Jagadish hereinafter) (US Patent No. 7,010,522 B1, filed: June 17, 2002).

Regarding Claim 7, the combination of Reiner in view of Koskas discloses all the limitations as disclosed above including length of predicate intervals (Col. 64, lines 32 – 35, Reiner) and inserting portions of predicate intervals into virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner). However, the combination of Reiner in view of Koskas is silent with respect to a method that determines if a predicate is larger than any of the virtual construct intervals. On the other hand, Jagadish discloses a method for inserting predicate intervals that comprises: inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 5, lines 22 – 27 and 37 – 39, Jagadish[32]), wherein a length in excess of a length of said initial portion of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[33]); and inserting the remnant predicate interval into the same length virtual

[32] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.
[33] The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish). It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the Jagadish's teachings to the system of the combination of Reiner in view of Koskas. Skilled artisan would have been motivated to do so, as suggested by Jagadish (Col. 2, lines 6 –7 and 29 – 31, Jagadish), to be able to efficiently find all strings approximately containing a given query substring from a large collection of strings by using all (or part) of overlapping pieces. In addition, the applied references (Reiner, Koskas, and Jagadish) teach features that are directed to analogous art and they are directed to the same field of endeavor, such as, databases management systems and indexing queries. This close relation between the applied references highly suggests an expectation of success.

Regarding Claim 14, the combination of Reiner in view of Koskas and further in view of Jagadish discloses a method, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish[34]), wherein a length in excess of a length of said initial portion of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[35]); and

[34] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.
[35] The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 21, the combination of Reiner in view of Koskas and further in view of Jagadish discloses a method, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish[36]), wherein a length in excess of a length of said initial portion of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[37]); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 28, the combination of Reiner in view of Koskas and further in view of Jagadish discloses a method, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 –

[36] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.
[37] The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

27 and 37 – 39, Jagadish[38]), wherein a length in excess of a length of said initial portion of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[39]); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 35, the combination of Reiner in view of Koskas and further in view of Jagadish discloses a service, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish[40]), wherein a length in excess of a length of said initial portion of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[41]); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

---

[38] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.
[39] The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.
[40] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.
[41] The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

Regarding Claim 42, the combination of Reiner in view of Koskas and further in view of Jagadish discloses a system, wherein if a predicate interval is larger than any of said virtual construct intervals, said predicate insertion handler:

inserts an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish[42]), wherein a length in excess of a length of said initial portion of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[43]); and

inserts the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 49, the combination of Reiner in view of Koskas and further in view of Jagadish discloses a program storage device, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish[44]), wherein a length in excess of a length of said initial portion

---

[42] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.
[43] The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.
[44] Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish[45]); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Response to Arguments

1. Applicant argues that the prior art fails to; "establish prima facie case of anticipation", because "the same exact feature of Reiner of "generating subqueries" is cited in the Office Action as disclosing both the "defining" and "inserting" features of the present invention".

Examiner respectfully disagrees. The applied art does disclose the claimed invention. Furthermore, as previously stated in the Final Office Action dated July 26, 2006, the citations presented are different (See Claim 1 rejection discussed in this Office Action above).

[45] The step of truncating the piece of query of length longer than q, implies that there is excess length of

Prior Art Made Of Record

1. Reiner et al. (US Patent No. 6,289,334 B1, issued: September 11, 2001) discloses an apparatus and method for recomposing database queries for database management system including multiprocessor digital data processing system.

2. Jagadish et al. (US Patent No. 7,010,522 B1, filed: June 17, 2002) discloses a method of performing approximate substring indexing.

3. Antoshenkov (US Patent No. 5,664,172) discloses a range-based query optimizer.

4. Koskas (US Patent No. 6,711,563 B1, filed March 5, 2001).

---

the predicates.

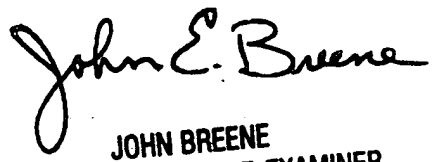
Points Of Contact

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Giovanna Colan whose telephone number is (571) 272-2752. The examiner can normally be reached on 8:30 am - 5:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Giovanna Colan
Examiner
Art Unit 2162
December 20, 2006



JOHN BREENE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100